Ciencia e Ingeniería
Revista Interdisciplinar de Estudios en Ciencias Básicas e Ingenierías.
Año 2014, Julio-Diciembre, Vol. (1) N° (2) ISSN 2389-9484.
Universidad de La Guajira, Facultades de Ciencias Básicas y Aplicadas e Ingeniería.
La Guajira-Colombia.
Revista en Línea http://revistas.uniguajira.edu.co/index.php/cei

MODELADO MATEMÁTICO DE BRAZO ROBOT CON 3DOF USANDO MATLAB

MATHEMATICAL MODELING OF A ROBOTICS MANIPULATOR WITH 3 DOF USING MATLAB

Sergio Velásquez, Jesús Luces

Centro de Investigación de las Redes Neuronales y la Robótica
CIRNAR-UNEXPO, Ciudad Guayana, Bolívar 8050 Venezuela.
svelasquez@unexpo.edu.ve

Recibido: Junio11 de 2014. Aceptado: Octubre 20 de 2014

RESUMEN

En el siguiente artículo se estudiara y desarrollara el proceso necesario para la realización del modelado matemático de un manipulador robótico antropomorfo con tres grados de libertad, partiendo de un proceso de naturaleza analítica como lo es, la parametrización de Denavit-Hartenberg que sintetiza en una tabla algunos parámetros que permiten expresar de manera estandarizada la geometría de brazos y manipuladores robóticos en general. Esta tabla, conocida como tabla de parámetros D-H, es la información inicial en la elaboración de modelados, tanto si el proceso a seguir es analítico, como computacional. En este apartado, se desarrollada con ayuda del toolbox «Damarob» instalado en el software base Matlab, calculándose el modelo cinemático directo, mediante el cual será posible visualizar el movimiento del manipulador en simulación, además de obtener los datos analíticos a través de matrices simbólicas. Una vez realizado la caracterización del modelo, la realización de las simulaciones permitirá entonces establecer conclusiones.

Palabras clave: Matlab, Robótica, Manipuladores, Modelado, Modelo.

ABSTRACT

In this article we study and develop the necessary process for the making of mathematical modeling about a robotic anthropomorphous manipulator with three grades of degrees, starting of a process of analytical nature like it is the parameterization of Denavit-Hartenberg which synthetizes in a data table some parameters that allow express of standard manner the geometry of the robotic manipulators in general. This table, is knows it like “parameters D-H table”, is the start information for modeling, whether the process to follow is analytical, as computational. In this section, it is develop with support of «Damarob toolbox» installed in the software base Matlab, calculates the kinematics direct model, through which was possible to see the movement of manipulator in simulation, in addition to obtain the analytical data through symbolical matrices. Once realizing the characterization of the model is created, with the carrying to the simulations was possible establish conclusions

Key words: Matlab, Robots, Robótica, Simulación, Modelado, Manipulador, Robotics, Manipulators.

I. INTRODUCCION

La historia de la robótica se origina desde la época antigua, incluso antes de haber sido nombrada y reconocida como ciencia. Esta historia empezaron a escribirla los egipcios y los griegos, quienes motivados por su fervor, se las ingeniaron para construir grandiosos aparatos para clamar a sus dioses y fascinar a los adoradores. Por un lado, los egipcios unieron eslabones rígidos de sus estatuas a través de articulaciones, mientras que por el otro, los griegos operaban las suyas con sistemas hidráulicos (Luces ,2015; Barrientos *et al* 2007).

Eslabones, articulaciones y sistemas hidráulicos son conceptos de desmedida importancia actualmente, en el diseño y desarrollo de manipuladores robóticos, estando ya consagrada la robótica como rama de la tecnología con extraordinarios avances como; la realización de trabajos forzosos, operaciones medicas de gran precisión e innumerables actividades de entretenimiento entre otros. De esta manera, en búsqueda de más avances en la rama de la robótica que se encuentra en sus mayores auges, surgen innumerables herramientas como paquetes de software para el estudio, diseño, simulación, pruebas de rigor y detección de fallos (Aguirre *et al*, 2011; Aguirre *et al*, 2009).

II. DESARROLLO

El proceso de modelado de cualquier brazo robótico, puede ser realizado de manera analítica o utilizando herramientas computacionales, sin embargo la manera de iniciarlo, sea cual sea el camino elegido para la realización del modelo, debe tener un inicio analítico. Este proceso analítico, se hace a través de la parametrización de Denavit–Hartenberg (Piñero y Velásquez, 2010).

Con este proceso, se obtiene una tabla con parámetros que describen la geometría de cualquier manipulador robótico Figura 1. El algoritmo de Denavit–Hartenberg (Albites *et al.* 2010), se hace de la siguiente manera:

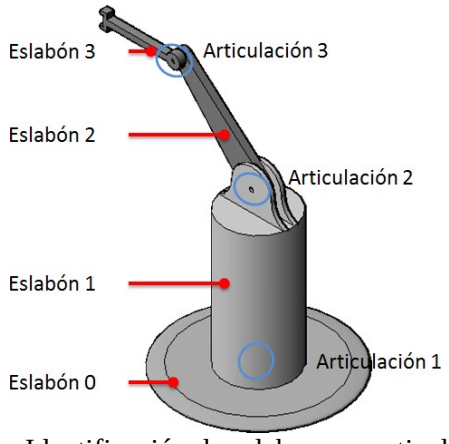


Figura 1. Identificación de eslabones y articulaciones

DH1: Numerar los eslabones comenzando con 1 (primer eslabón móvil de la cadena) y acabando con $\boldsymbol{n}$ (último eslabón móvil). Se numerará como eslabón 0 a la base fija del robot Figura 2.

DH2: Numerar cada articulación comenzando por 1 (correspondiente al primer grado de libertad) y acabando en $\boldsymbol{n}$.

DH3: Localizar el eje de cada articulación. Si ésta es rotativa, el eje será su propio eje de giro. Si es prismática, será el eje a lo largo del cual se produce el desplazamiento.

DH4: Para $\boldsymbol{i = 0}$ hasta $\boldsymbol{n - 1}$ situar el eje $\boldsymbol{Z_i}$ sobre el eje de la articulación $\boldsymbol{i + 1}$.

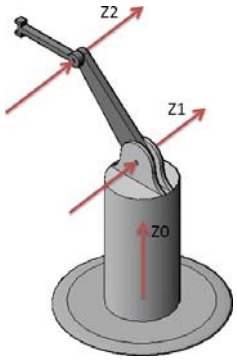


Figura 2. Ejes de articulación y ejes de giro.

DH5: Situar el origen del sistema de la base $\{\boldsymbol{S_0}\}$ en cualquier punto del eje $\boldsymbol{Z_0}$. Los ejes $\boldsymbol{X_0}$e $\boldsymbol{Y_0}$se situarán de modo que formen un sistema dextrógiro con $\boldsymbol{Z_0}$.

DH6: Para $\boldsymbol{i = 1}$ hasta$\boldsymbol{n - 1}$, situar el sistema $\{\boldsymbol{S_i}\}$ en la intersección del eje $\boldsymbol{Z_i}$con la línea normal común a$\boldsymbol{Z_{i-1}}$ y $\boldsymbol{Z_i}$. Si ambos ejes se cortasen se situaría $\{\boldsymbol{S_i}\}$ en el punto de corte. Si fuesen paralelos $\{\boldsymbol{S_i}\}$ se situaría en la articulación $\boldsymbol{i + 1.}$ Figura 3

DH7: Situar $\boldsymbol{X_i}$ en la línea normal común a $\boldsymbol{Z_{i-1}}$ y $\boldsymbol{Z_i}$.

DH8: Situar $\boldsymbol{Y_i}$ de modo que forme un sistema dextrógiro con $\boldsymbol{X_i}$ y $\boldsymbol{Z_i}$.

DH9: Situar el sistema $\{\boldsymbol{S_n}\}$ en el extremo del robot de modo que$\boldsymbol{Z_n}$coincida con la dirección de $\boldsymbol{Z_{n-1}}$ y $\boldsymbol{X_n}$sea normal a $\boldsymbol{Z_{n-1}}$y $\boldsymbol{Z_n}$.

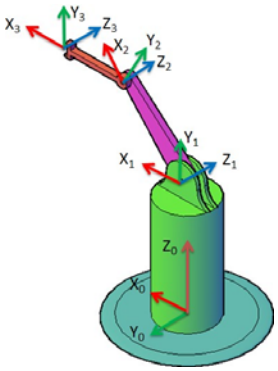


Figura 3. Brazo robótico referenciado.

DH10: obtener $\boldsymbol{\theta_i}$ como el ángulo que hay que girar en torno a $\boldsymbol{Z_{i-1}}$ para que $\boldsymbol{X_{i-1}}$ y $\boldsymbol{X_i}$ queden paralelos. Se trata de un parámetro variable en articulaciones giratorias.

DH12: obtener $\boldsymbol{a_i}$ como la distancia a lo largo del eje $\boldsymbol{X_i}$, que va desde la intersección del eje $\boldsymbol{Z_{i-1}}$ con el eje $\boldsymbol{X_i}$ hasta el origen del sistema$\boldsymbol{i-esimo}$, en el caso de articulaciones giratorias. En el caso de articulaciones prismáticas, se calcula como la distancia más corta entre los ejes $\boldsymbol{Z_{i-1}}$y $\boldsymbol{Z_i}$.

DH13: obtener $\boldsymbol{\alpha_i}$ como el ángulo de separación del eje $\boldsymbol{Z_{i-1}}$ y el eje $\boldsymbol{Z_i}$, medido en un plano perpendicular al eje $\boldsymbol{X_i}$, utilizando la regla de la mano derecha.

Tabla 1. Parámetros D-H

Articulación	$\boldsymbol{\theta}$	$\boldsymbol{d}$	$\boldsymbol{a}$	$\boldsymbol{\alpha}$
1	θ_1	L_1	0	90°
2	θ_2	0	L_2	0
3	θ_3	0	L_3	0

Con la 0, obtenemos la información necesaria para traducir la configuración geométrica del brazo manipulador al software que se utilizara para el modelado, que en este caso será Matlab.

Matlab ofrece la posibilidad de trabajo mediante toolbox, que como lo dice su nombre “Caja de Herramientas”, con paquetes que facilitan actividades en áreas específicas como el modelado robótico. En la red, es posible conseguir muchos de estos toolbox, desarrollado con programadores con experiencia, que lo ofrecen de manera gratuita y libre para uso personal (Caputano y Bellicoso, 2009).

Este es el caso de Dario Bellicoso y Marco Caputano, especialistas en control y mecatrónica, quienes desarrollaron un toolbox para del modelado cinemático, dinámico de manipuladores robóticos a través de matrices simbólicas. Este toolbox se puede descargar de manera gratuita a través de su página web en (Craig, 2006; Caputano y Bellicoso, 2009).

El toolbox «Damarob» se debe instalar siguiendo descomprimir el archivo de cual se derivara una carpeta llamada Figura 4.

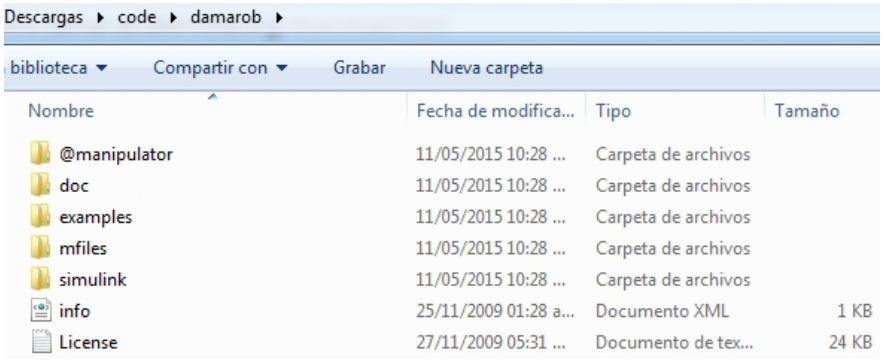


Figura 4. Subcarpetas del Toolbox

Posteriormente ejecutar las siguientes líneas de comandos, con la ubicación de la carpeta «damarob» y sus respectivas subcarpetas:

$$path(path,'C:\Program\ Files\MATLAB\R2014b \toolbox\damarob')$$

$$path(path,'C:\Program\ Files\MATLAB\R2014b \toolbox\damarob\doc')$$

$$path(path,'C:\Program\ Files\MATLAB\R2014b \toolbox\damarob\examples')$$

$$path(path,'C:\Program\ Files\MATLAB\R2014b \toolbox\damarob\files')$$

$$savepath$$

$$dama_install$$

Al finalizar la instalación del toolbox, un mensaje aparecerá indicando que el proceso se realizó de manera satisfactoria, por lo que se podrán acceder a las funciones que este trae.

La clave que da inicio al modelado, son los datos contenidos en 0, pues de hecho, son el punto de partida para la realización de cualquier modelado de brazo robótico. De tal manera que es posible usar la función «$manipulator$». Para poder invocar esta función, se debe escribir en la ventana de comandos de Matlab: $Manipulator(DH, name, Tb0, Tne)$ CITATION FuK \l 3082 (González et al. 2007).

En donde:

DH: Es la tabla de parámetros D-H, que incluye el tipo de articulaciones y sus límites. Debe ser una matriz de dimensión $nx7$. Las columnas n, es el número de articulaciones o enlaces, mientras que las filas guardaran los datos en este orden:

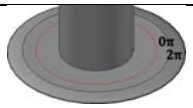
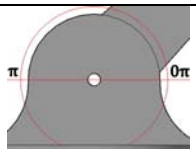
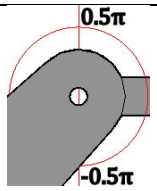
1. Distancia a_i
2. Ángulo α_i
3. Distancia d_i
4. Ángulo θ_i
5. R/P, es una bandera para indicar si la articulación es $revoluta = 0, prismática != 1$
6. Min y Max, determinan el rango de movimiento de las articulaciones, sea revoluta o prismática. De no indicarse, entonces asumir como rango$(-\infty, +\infty)$.

$name$: Es una cadena de caracteres donde se le da nombre al manipulador. El nombre debe ir entre comillas simples (‘’) y no puede empezar por un número.

Las matrices $Tb0$ y Tne, representan la ubicación espacial de la base y el efector final respectivamente. Si no se especifican, asumen el valor de la matriz identidad.

Conociendo los parámetros de entrada de la función y lo que significan cada uno, se puede proceder a generar el modelo 3D, no sin antes complementar la información de la tabla 1, con la delimitación que deseamos o requerimos establecer en cada uno de los ejes de articulación de nuestro brazo manipulador. **¡Error! No se encuentra el origen de la referencia.**

Tabla 2. Límites de Articulación

Articulación	Espacio de Trabajo	Rango
$\boldsymbol{\theta_1}$		$[0,2\pi]$
$\boldsymbol{\theta_2}$		$[0,\pi]$
$\boldsymbol{\theta_3}$		$\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$

Así, las consideraciones finales para invocar la función «*manipulator*», se pueden observar en Tabla 3

Tabla 3. Parámetros de entrada función «*manipulator*».

Variable	Código Matlab	Justificación
$\boldsymbol{A_i}$	>> Ai = [1,0,0,0; 0,1,0,0; 0,0,1,0; 0,0,0,1]	Se asumirá la posición básica de la base.
$\boldsymbol{A_f}$	>> Af = [1,0,0,0; 0,1,0,0; 0,0,1,0; 0,0,0,1]	Se asumirá la posición básica del efecto final.
$\boldsymbol{DH}$	>> DH = [0, pi/2, 1, 0, 0, 0,2*pi; 0.7, 0, 0, 0, 0, 0,pi; 0.4, 0, 0, 0, 0, -pi/2,pi/2]	Para los longitudes los valores serán: $l_1 = 1.0$ $l_2 = 0.5$ $l_3 = 0.5$

Antes de visualizar cada modelo, es necesario tener en cuentas las siguientes consideraciones:

1. Los ángulos en Matlab se trabajan siempre en radianes, usando el monosílabo "*pi*".
2. La función «*manipulator*» genera un modelo 3D que es posible guardar en una variable cualquiera, aspecto que será útil si se quiere caracterizar en un bloque más adelante.
3. Al generar un modelo 3D, se generan algunos archivos que no desaparecen al cerrar Matlab, o borrar todas las variables. De tal forma que se debe borrar manualmente, para evitar conflictos en el futuro. Figura 5.

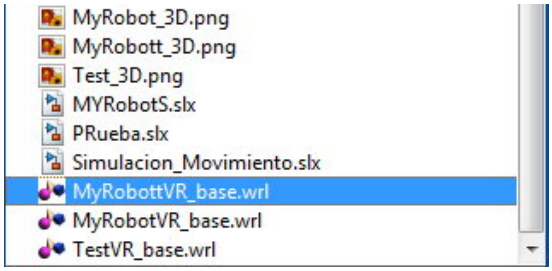


Figura 5. Archivo de Modelado 3D .wrl

III. RESULTADOS Y DISCUSIÓN

Como resultado al ejecutar la siguiente línea de comando:

$$v_robot = Manipulator(DH, name, Tb0, Tne)$$

Se obtiene el modelo 3D de un robot con 3DoF, de la figura a continuación:

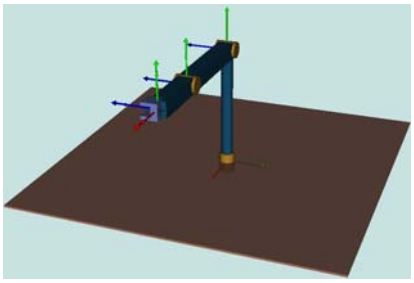


Figura 6. Modelo 3D del manipulador robotico

No obstante, no es posible probar el movimiento articular en esa ventana, razón por la cual hacemos entonces, uso de cualquiera de estas dos sintaxis, que corresponden a la misma función (Jaramillo, 2010):

$drive.manip(v_robot)$.

$v_robot.drive_manip$.

Ejecutando cualquier de las anteriores opciones, ahora disponemos de una ventana que permite el manejo de las articulaciones a través de cursores de dirección. Figura 7.

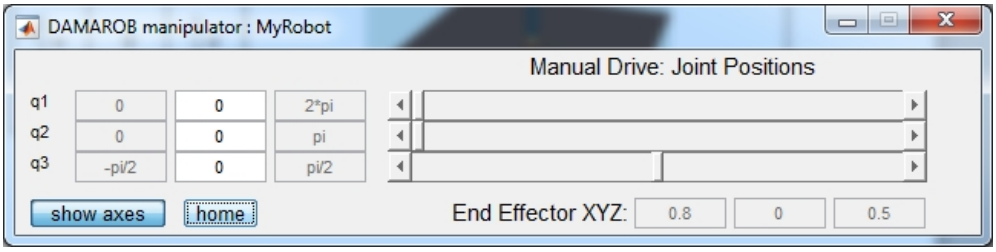


Figura 7. Ventana de Manejo

$drive.manip(robot)$, puede funcionar de dos maneras: la primera haciendo uso de los cursores que permitirán el desplazamiento de las articulaciones en los límites establecidos para cada articulación o la segunda, introduciendo los valores de los ángulos en radianes en loa recuerdos blancos.

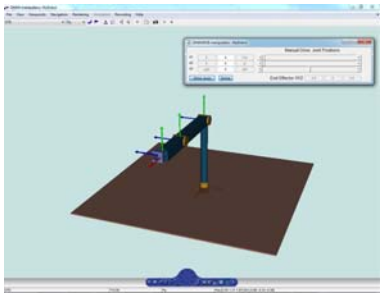
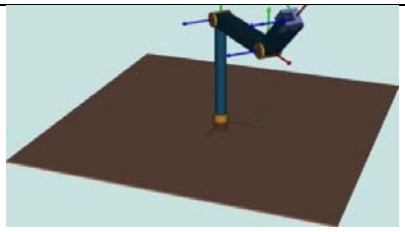
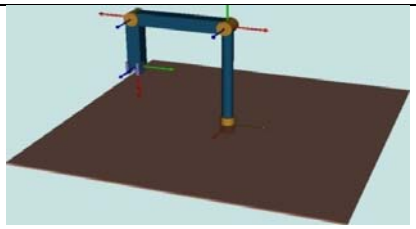
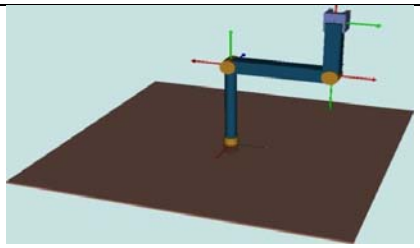
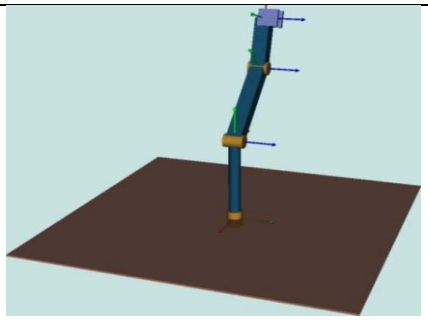
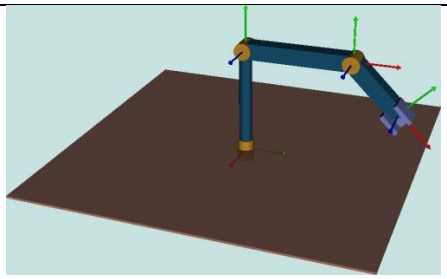


Figura 8. Modelo 3D con ventana de Manejo

Aprovechando entonces la funcionalidad de posicionamiento directo, se procedió hacer pruebas introduciendo ángulos notables, en el rango de cada articulación. 0

Tabla 4. Prueba de posicionamiento con la función «drive»

$\boldsymbol{\theta_1}$	$\boldsymbol{\theta_2}$	$\boldsymbol{\theta_3}$	Posición
$\boldsymbol{\frac{1}{4}\pi}$	0	$\frac{1}{4}\pi$	
$\boldsymbol{\frac{1}{2}\pi}$	π	$\frac{1}{2}\pi$	
$\boldsymbol{\frac{3}{2}\pi}$	π	$-\frac{1}{2}\pi$	

$\boldsymbol{\pi}$	$\frac{1}{4}\pi$	$\frac{1}{6}\pi$	
$\boldsymbol{\frac{1}{2}\pi}$	0	$-\frac{1}{4}\pi$	

IV. CONCLUSIONES

El diseño completo de un robot o manipulador robótico es un área de trabajo que necesita del conocimiento de diferentes áreas como la mecánica, la electrónica, informativa y matemática, haciéndose un proyecto que debe ser interdisciplinario, para poderlo llevar a cabo satisfactoriamente.

Para la realización de otro tipo de modelado, como el dinámico, que considera las causas que genera el movimiento, es necesario obtener información más completa en el diseño de los brazos robóticos, como la información de los motores, sus rotores, la dimensión exacta de los eslabones, el material del mismo y el peso.

Matlab con sus toolbox y herramientas es un programa ideal para la realización de estos modelos, ya que su base de trabajo son las matrices, necesarias para el posicionamiento, rotación y ubicación de un cuerpo en el espacio. Además de esto, la probabilidad de programar algoritmos o descargar otros hechos por programadores anónimos que facilitan el análisis que analíticamente corresponde a un trabajo muy difícil y engorroso.

Otras de las ventajas de Matlab, el hecho de no tener limitaciones entre el entorno escrito como la ventana de comandos y entornos gráficos como simulink, ya que existen bloques como el interpretador de funciones de Matlab y otros, que pueden vincular los dos entornos cuando es necesario.

V. REFERENCIAS BIBLIOGRAFICAS

Aguirre Gil, L. Andueza y C. Arismendi, 2011. Sistema manipulador antropomórfico de tres grado libertad, Madrid,

Aguirre Gil, C. César y L. Andueza, 2009. Sistema manipulador antropomórfico de tres, Madrid,

Albites, M. Rimachi y J. Paucar, 2010.«Diseño y Análisis de Controladores para un robot de 3GDL con capacidad de dibujo,»

Barrientos, L. Peñin y R. Aracil, 2007. Fundamentos de Robotica, Madrid: McGraw,

Caputano y D. Bellicoso, 2009. «DAMAROB Dario and Marco's Robotics Symbolic Toolbox for Matlab,» Noviembre [En línea]. Available: http://www.damarob.altervista.org/aboutus.html. [Último acceso: 7 6 2015].

Craig J. 2006., Introducción a la Robótica, México: Pearson

González Rodríguez, M. Pineda Ortega y D. M. Soberanes Leal, 2007.Seguimiento Adaptativo de Trayectorias con Convergencia en el Tiempo Finito de un Robot Antropomórfico Virtual de Tres Grados de Libertad, Pachuca de Soto, Hidalgo: Universidad Autónoma de Estado de Hidalgo,

Jaramillo Rojas, 2010. Modelo Simulación y Control de un Brazo Robotico Mediante Matlab y Simulink para Soldadura de Arco, Quito,

Luces Gil, «Modelado y simulación de un brazo robot con 3dof y control usando Matlab.,» Puerto Ordaz, 2015.

Piñero y S. Velásquez , 2010. «Control de Brazo Robotico,» *Memorias de las IX Jornadas de Investigación - UNEXPO Puerto Ordaz,*

Ramírez Benavides, «Escuela de Ciencias de la Computación e Informática,» [En línea]. Available: http://www.kramirez.net/.

.